IJTRET "Voice of Innovative Minds"

doi.org/10.54473/IJTRET.2021.5107

International Journal of Trendy Research in Engineering and Technology
Volume 5 Issue 1 February 2021
ISSN NO 2582-0958

Aerial pictures semantic segmentation applying deep learning

Abhishek Solanki, Rajan Kumar Singh, Brinsley Demenezes
Department of Computer Engineering, Xavier Institute of Engineering, Mumbai University, Mumbai, India

ABSTRACT

An obvious expansion in the measure of satellite dataset accessible lately has made the translation of this information a difficult issue at scale. Determining helpful insights from such pictures requires a rich comprehension of the data present in them. AI is currently utilized for keeping up precise automated regional maps to react to real time, natural and catastrophe recuperation challenges. These assignments need close to continuous, precise, mechanized planning straight from aerial and satellite pictures. In this project, we apply Mask-RCNN and Conditional Adversarial Network techniques for extracting building footprint. The problem is viewed as a supervised learning problem. We try different things with learning parameters and algorithms, apply data augmentation, use transfer learning, utilizing RGB data and to accomplish high precision results. The resulting pipeline incorporates image pre-processing algorithms that permits it to adapt to input pictures of fluctuating quality, resolution and channels.

Keywords: Mask-RCNN, Conditional Adversarial Networks, deep learning, image segmentation, satellite imagery

I. INTRODUCTION

Significant manual work is still required to produce accurate quality maps of a region. Automating this task is a very interesting and difficult task. In this paper we explore state of the art Mask R-CNN method for instance segmentation, and Conditional Adversarial Network image-to-image translation for semantic segmentation. The main aim of doing this is to build a Computer Vision Web application that extracts and prints building footprint from given high resolution satellite image of a region using deep learning and, in the process, implement and experimentally evaluate deep neural networks for the semantic segmentation of satellite images.

One can think of the Mask R-CNN as a mask generator. We can then take these generations and optimize for making them more accurate by using the minmax strategy used in GANs where there is another network – discriminator – that tries to distinguish them from the real ground truth masks. We show this strategy outperforms the hand-engineered losses. Satellite imagery is usually of high resolution and takes a lot of space. So, it is very important to precisely preprocess the dataset before giving it to the model. This project is designed to implement a deep neural network that takes a satellite image and extracts the building footprint from it. Understanding the urban growth and infrastructure expansion is highly correlated with building, roads and highways. The resulting pipeline shall include image pre-processing algorithms to cope with input images of varying quality, resolution, and channels. This type of application would be helpful for generating quick and automatic maps of areas which can help in city planning, disaster rescue operations, for research studies etc.

II. RELATED WORK

Understanding a picture and classifying its content into semantic groups translates into formulating a per-pixel classifier, where we predict a category for each pixel within the image, and extract a semantic map of the whole image. The same idea is often extrapolated into serving for a multi-class classification, where one would consider semantic groups like buildings, meadows and rivers. The following techniques were mostly used in the research papers we referred. Elliott et al [1] have used a Fully Convolutional Neural Net- work to extract bounding polygons for building footprints. They used the dataset first released as part of the first Space Net Challenge. The winning implementation produced an F1 score of and used no deep learning. They used deep learning and were able to achieve an F1 score of 0.34 which outperformed the previous one. We believe to train the FCNN on top of a pretrained CNN such as VGG would have been much feasible. The plus point being that the VGG model has learned general features from a larger image corpus, and then we can fine-tune the model to the task of satellite image segmentation with our smaller Space Net dataset. Approaching building footprint extraction as an instance segmentation problem would have been an upgrade. We can also simplify the pipeline by directly extracting bounding polygons using a Mask RCNN, rather than first predicting a heatmap and then applying Marching Squares.

The first paper where possibility of using a U-Net architecture for building footprint extraction was discussed was Ternaus Net: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation [2].UNET is capable of learning from a relatively small training set. In most cases, data sets for picture segmentation consist of at most thousands of pictures, since manual preparation of the masks is a very costly procedure. Usually U-Net is trained from scratch starting with randomly initialized weights.

In ref. [7] the authors develop a U-Net specifically dedicated to biomedical picture segmentation achieving accuracy of about 75 the test sets.

In [2] the authors have successfully trained an UNet architecture to remote sensing imagery.

The difference of this approach to FCN4s and FCN8s is that it uses more skip connections. Picture resolution is recovered at the last skip connection. Although Iglovikov et al [2] have achieved high-quality results, they do not utilize enough of the available data sources, which leaves potential for improvement. Drawbacks of [2] was that the loss function could not take into consideration the precision on the boundary along with the accuracy on the segmentation task. So, a better loss function was needed so as to output a better IOU.

III. III DESCRIPTION

Understanding an image and classifying its content into semantic groups translates into formulating per-pixel classifier, where we predict a class for each pixel in the image, and extract a semantic map of the entire image. The same idea can be extrapolated into serving for a multi- class classification, where one would consider semantic groups such as buildings, meadows and rivers. But here we majorly focus on segmentation for the purpose of generating building maps. Building footprints can help one to identify the following information 1. Number of buildings within a given perimeter or a property 2. Risk identification from nearby trees, water, and other hazardous surrounding elements. 3. Complete locational accuracy by verifying a building's exact location through rooftop geofencing methods .4. The building geometry data can be used to associate other relevant data that may relate to a specific location, business, product, market, etc.

The cost of the project depends upon the training of the model and system requirements for that purpose such as CPU RAM, GPU and disk space. A computer with about 8 GB ram and basic GPU card was used in unison with shared CPU and GPU processing offered by Google Colab. The disk space offered by the same was also used.

Hence, by cost-benefit analysis we can conclude that benefit to cost ratio is high. The user just has to upload satellite image for the region. The image just has to be of high quality which can be easily taken from google maps or other opensource aps.

Hence, the controls are simple and basic. Efficiency of the project is based upon few factors, namely, the model and its individual accuracy, the front end and its latency, the inter-connectivity. All these components can again be controlled individually and hence accuracy of the project as a whole can be easily controlled.

A. Design- Block Diagram

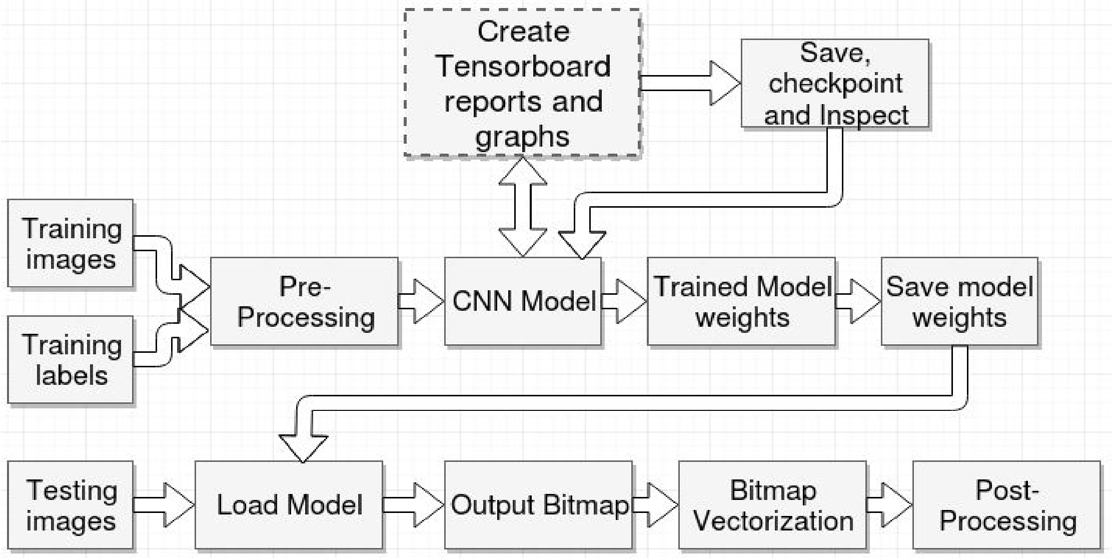


Figure 3.1: Block Diagram

The design for the proposed system is shown in the Figure 3.1. The images and their masked labels are in .tiff format. The training images are pre-processed before passing them to the model. The training images and labels are then fed to the CNN model.

TensorFlow provides graphs for monitoring the accuracy and loss while training. After training, the model is saved in HDF5 format. An area is selected for testing the trained model. The prediction is performed on the selected area by loading the trained model which outputs a bitmap.

It is then converted into a vector format and saved in as a shape file. In order to improve the shape of the building polygons and remove the noise after prediction, some post-processing algorithms are applied.Figure 3.2 shows the activity diagram of the proposed system. The user has to upload a satellite image of a region and upload it on the website. After passing quality check the model runs on the image and produces masked segmented image classifying the picture into building and non-building classes. Then it outputs the newly generated segmented image along with the original picture.

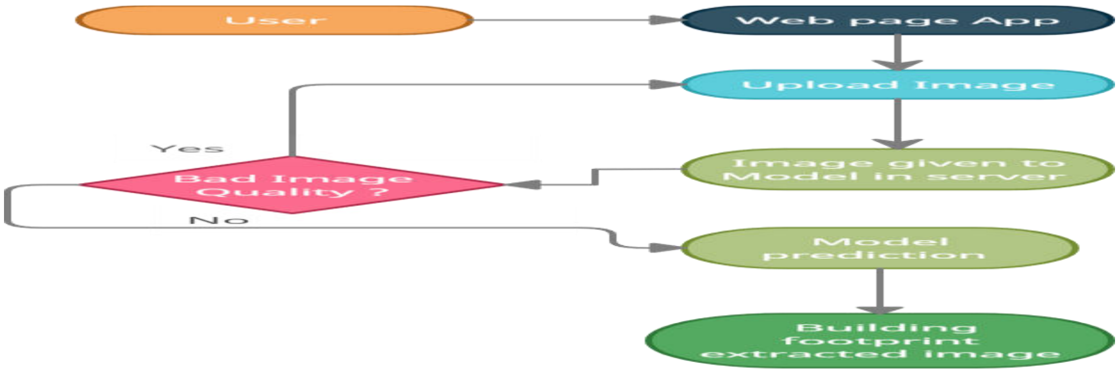


Fig 3.2 Activity Diagram

IV. IMPLEMENTATION METHODOLOGY

This context dwells on the detailed pipeline of the semantic segmentation. The first section briefs us the idea behind the computational graphs, and the next one will dive into TensorFlow as a deep learning framework. The best way to visualize the equations is to represent them as graph networks, where every computation forming a node in the graph. We've already gone through one example in the above section where we introduced about back-propagation. A core concept in functional programming, the back-propagation algorithm can be intuitively understood with this representation.

In a Deep neural network, computational graphs can be really complex, as it contains at least a million parameters. To build a Neural-Network from scratch is a terrifying task, with many mathematical and implementation specific subtleties to be addressed. For this reason, we generally make use of the frameworks that take out most of the computations away from the user.

A. Mask R-CNN

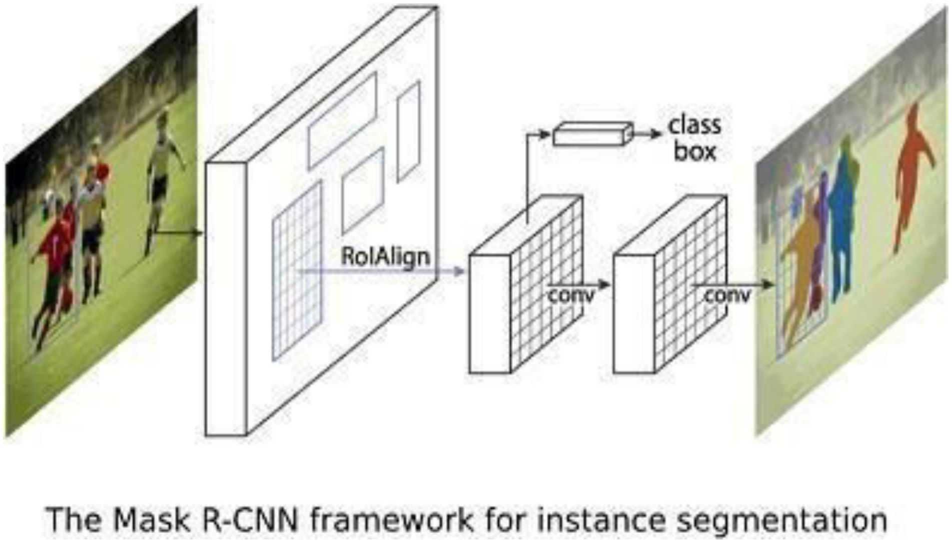


Figure 3.4: Mask R-CNN

Mask R-CNN architecture is state-of-the art approach for instance segmentation. The model consists of:

- Backbone CNN for feature extraction
- Region Proposal Network to find anchor boxes
- Segmentation Masks to identify objects masks

As we can see in Figure 3.4, the model receives input image, it extracts features from it using multiple convolutional/Batch Norm/MaxPool layers from the backbone network.

The next stage RPN uses the extracted features and anchors of different sizes and shapes to find proposals for foreground objects and for their approximate bounding boxes. The next stage ROI Classifier assigns proposed objects to one of the output classes (which is only one class in our case) and refines their bounding boxes. The last step Segmentation Mask applies multiple convolutional layers on top of proposals to generate object masks.

B. Conditional Adversarial Network

Conditional Adversarial Network combines GAN objective with traditional L1 loss. The discriminator is unchanged, except it classifies separately each of N × N (where N is a hyper-parameter) grid patches from last convolutional layer. Then each output grid patch outputs if it is real building footprint patch or generated. Figure 3.5 demonstrates basic implementation of a GAN model.

Generative modeling is an unsupervised learning task in machine learning that involves automatically discovering and learning the regularities or patterns in input data in such a way that the model can be used to generate or output new examples that plausibly could have been drawn from the original dataset.

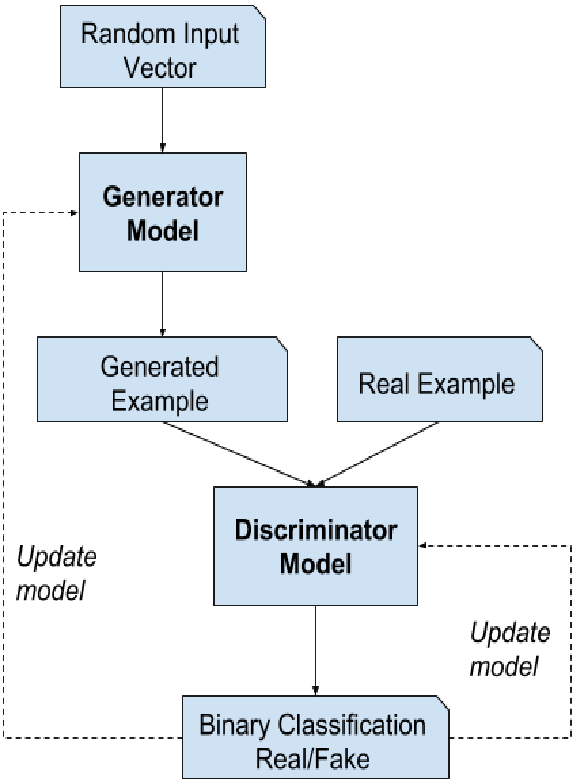


Figure 3.5: GAN model

V DETAILS OF HARDWARE AND SOFTWARE

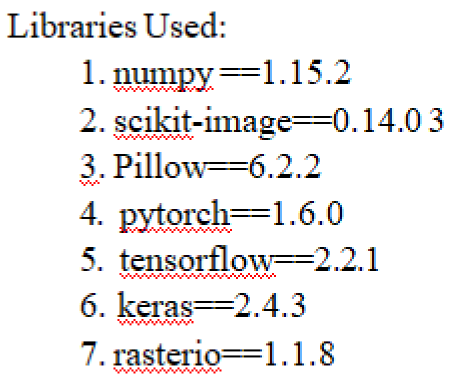
To ensure the entire project runs smoothly we would need to match certain hardware and software requirements.

Hardware Requirements:

1. A computer with at least 8GB RAM to ensure smooth executing of different processes.
2. A Nvidia graphic card (GPU) can be used to drastically speed up the entire execution time. We used NVIDIA GEFORCE GTX 1650 4GB.

Software Requirements

1. Python 3 programming language has to be installed.
2. Anaconda navigator and anaconda prompt has to be installed.
3. Django and React



Libraries Used:

1. numpy ==1.15.2
2. scikit-image==0.14.0 3
3. Pillow==6.2.2
4. pytorch==1.6.0
5. tensorflow==2.2.1
6. keras==2.4.3
7. rasterio==1.1.8

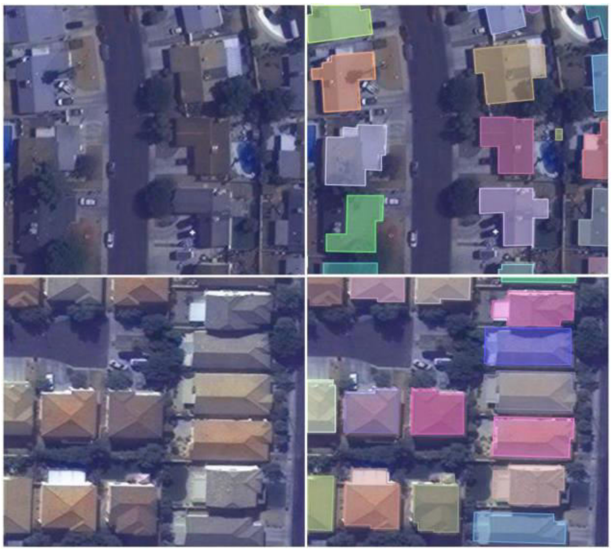


Fig.3.5 Output

VI RESULTS AND DISCUSSIONS

We presented an implementation of Mask R-CNN with a GAN loss. The idea is to make the baseline mask generator as the GAN generator, and then design a discriminator network to challenge these generations.

As a result of our venture, we are implementing 2 models that automatically map building footprints with the accuracy that is very close to human-level. The models that we will be using primarily are based on the state-of-the-art architectures Mask-RCNN and GAN. For reference the output of such a system is shown below.

We have researched and found that these models will totally fulfill our needs for the application. We will also do error analysis and compare to provide graphic visualization on the model results at various stages in model's pipeline.

VII CONCLUSION

A significant inspiration for undertaking this work was to gain an understanding for planning, evaluating and assessing a profound learning pipeline with the emphasis on satellite image information. Our method largely depends upon the training data provided and also the quality of the images in the training data. When provided with the training images from different countries the same model can be used in the future to make a system that can detect and extract buildings from any part of the world. We expect the experimental results to show that the proposed system can accurately and quickly extract the building footprints from satellite images with good accuracy.

The same pipeline can be further used for extracting other features like vegetation, roads, highways, water bodies etc. from satellite images. Training the pipeline on datasets from several cities around the globe will help the model in generating building footprint of satellite picture of any region on earth.

REFERENCES

1. Elliott Chartock ,Whitney LaRow, Vijay Singh :Extraction of Building Footprints from Satellite Imagery

2. Iglovikov, V.; Shvets, A. TernausNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation. arXiv 2018, arXiv:1801.05746.

3. M. Vakalopoulou, K. Karantzalos, N. Komodakis, N. Paragios, Building Detection in Very High Resolution Multispectral Data with Deep Learning Features, Nov 2015

4. Benjamin Bischke, Patrick Helber, Joachim Folz, Damian Borth, Andreas Dengel, Multi-Task Learning for Segmentation of Building Footprints with Deep Neural Networks,

arXiv:1709.05932v1 [cs.CV], September 2017.

5. Bic-User. Topcoder: bic-user's implementation. https://github.com/SpaceNetChallenge/Building De tree/master/bic- user, 2017. Github.

6. https://medium.com/geoalert-platform-urban-monitoring/buildings-height-estimation-7babe642089

7. O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, Computer Science Department and BIOSS Centre for Biological Signalling Studies, University of Freiburg, Germany, 2015.

8. M. Kampffmeyer, A.-B. Salberg, and R. Jenssen. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neuralnetworks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 1–9, 2016.

9. https://spacenet.ai/spacenet-buildings-dataset-v2/

10. https://towardsdatascience.com/semantic-segmentation-of-aerial-images-using-deep-learning-90fdf4ad780

11. https://developers.arcgis.com/python/sample-notebooks/automate-building-footprint-extraction-using-instance- segmentation